

Scientific journal  
**PHYSICAL AND MATHEMATICAL EDUCATION**  
 Has been issued since 2013.

Науковий журнал  
**ФІЗИКО-МАТЕМАТИЧНА ОСВІТА**  
 Видається з 2013.

ISSN 2413-158X (online)  
 ISSN 2413-1571 (print)



<http://fmo-journal.fizmatsspu.sumy.ua/>

Кучеренко І.А. Використання RENDERSCRIPT на Android-пристроях // Фізико-математична освіта : науковий журнал. – 2016. – Випуск 4(10). – С. 62-65.

Kucherenko I. Using Renderscript on Android-devices // Physical and Mathematical Education : scientific journal. – 2016. – Issue 4(10). – P. 62-65.

УДК 004.94+001:372.82

І.А. Кучеренко

Український державний хіміко-технологічний університет, Україна

### ВИКОРИСТАННЯ RENDERSCRIPT НА ANDROID-ПРИСТРОЯХ

**Вступ.** За даними International Data Corporation (IDC), міжнародної компанії маркетингових досліджень, консультаційних послуг, організатор конференцій в області інформаційних технологій, телекомунікацій і споживчої електроніки, світовий ринок смартфонів виріс на 0.7% у 2016 році. Пристрої під керуванням операційної систем (далі ОС) Android домінують за кількістю з часткою 87,6% (рис. 1.). Це зростання відбувається в першу чергу за рахунок розвинених ринків, таких як США, Канада, Японія і Західна Європа.

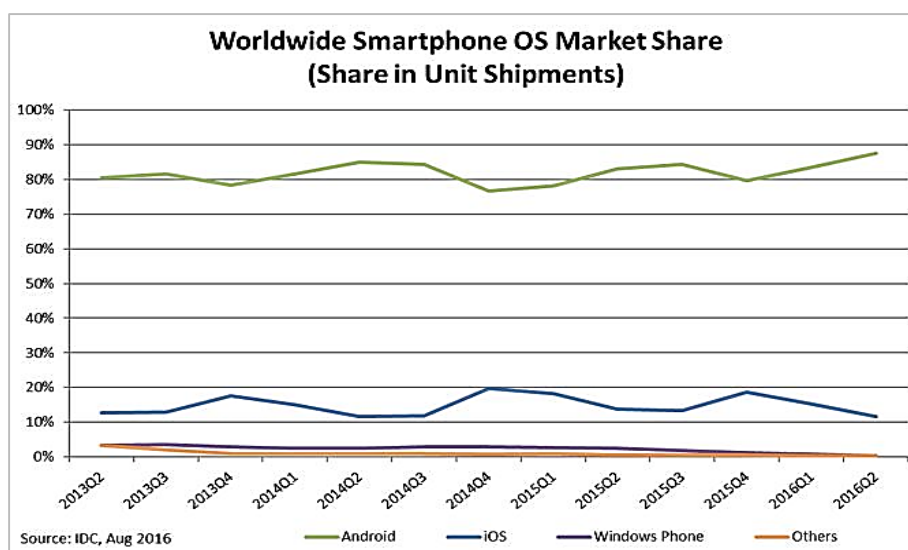


Рис. 1. Частки пристроїв з різними ОС

Смартфони (з англ. *smart* — розумний, і англ. *phone* — телефон) — окрема категорія телефонів, які — на відміну від простих стільникових телефонів — мають більше оперативної пам'яті і власний потужний, як для кишенькових пристроїв процесор, працюють під операційною системою Symbian 6.1 і вище, операційними системами платформи Windows Mobile 5 і вище або Palm OS, операційною системою iOS, Android, Bada. Завдяки таким даним підтримують багато програм написаних на C++ та java, зокрема 3D-ігри [1].

Смартфони відрізняються від звичайних телефонів наявністю розвинутою операційної системою, можливістю встановлювати додаткове програмне забезпечення, найчастіше від сторонніх розробників, для додавання нових можливостей. Розширення функціональності можливо за рахунок J2ME-програм, які підтримуються практично всіма мобільними телефонами, смартфонами і комунікаторами. Екран цілого ряду мобільних телефонів не поступається більшості смартфонів (останнім часом телефони оснащуються і сенсорними екранами), багато моделей оснащені роз'ємом для картки пам'яті. Смартфони в очах користувачів виглядають привабливішими за рахунок багатьох факторів (частково зумовлені просуванням смартфонів і комунікаторів виробниками), таких як: сучасні мультимедійні функції (якісніша камера, розширені можливості відтворення відеофайлів, поліпшені музичні здібності), Wi-Fi, GPS і інших.

Важливо відзначити, що програми, написані спеціально для операційної системи смартфона або комунікатора є повноцінними скопійованими в двійковий код послідовностями низькорівневих мікропроцесорних команд. А з урахуванням того, що у всіх смартфонах встановлені потужніші процесори, ніж у мобільних телефонах, можливості програміста зі створення таких програм практично не обмежені. Спеціалізовані програми функціональніші й раціональніше використовують ресурси процесора, ніж «універсальні» J2ME-програми. Тому смартфони користуються популярністю серед розробників програмного забезпечення і ентузіастів [2].

В даний час не існує чіткого розмежування між смартфонами і комунікаторами, оскільки функціональність обох класів пристроїв приблизно однакова. Різні експерти і виробники по-різному трактують ці терміни. Часто застосовується так званий «історичний підхід», який полягає в наступному: якщо пристрій веде свій родовід від КПК — то це комунікатор, а якщо від мобільних телефонів — то це смартфон. У рамках цього підходу під комунікаторами зазвичай маються на увазі пристрої з сенсорним екраном (може бути доповнений клавіатурою) працюють під управлінням операційної системи Windows Mobile. Пристрої з Windows Mobile, що використовують для введення інформації виключно QWERTY — і / або цифрову клавіатуру (аналог телефонного), називаються смартфонами. Більшість пристроїв під управлінням Symbian OS традиційно відносять до смартфонів (за винятком Nokia серій 9xxx, Nokia E90 і деяких інших). В інших випадках позиціонування пристрою залежить від виробника (зазвичай пристрої з сенсорним екраном відносять до комунікаторів, а до смартфонів відносять пристрої без такого екрану).

Користуватися смартфонами дуже зручно, адже вони дозволяють будь-де та будь-коли виконувати різноманітні операції: відправляти електронного листа, переглядати новини, створювати документи та навіть редагувати зображення чи відео. Проте розробникам мобільних додатків потрібно пам'ятати: перевагою ОС Android для таких пристроїв є націленість на пристрої з обмеженими ресурсами. Програми повинні виконувати свої функції швидко, не зловживаючи оперативною пам'яттю та запасом акумулятора.

Основною мовою програмування для ОС Android є Java. Це об'єктно-орієнтована мова програмування, яка компілюється у байт-код, який при виконанні інтерпретується віртуальною машиною для конкретної платформи. Передусім Java розроблялась так, щоб працювати незалежно від цільової платформи, тому вона має менше низькорівневих можливостей для роботи з апаратним забезпеченням, що у певних ситуаціях дало б значний приріст продуктивності. За необхідності таких дій Java дозволяє викликати підпрограми, написані іншими мовами програмування, наприклад: C, C++, Assembler, Render Script.

RenderScript — це прикладний програмний інтерфейс (API), який включає функції для 2D/3D рендеринга і математичних обчислень з високою продуктивністю. Він дозволяє описати будь-яку задачу з однотипними незалежними обчисленнями над великим об'ємом даних і розбити її на однорідні підзадачі, які можуть бути виконані швидко і паралельно на багатоядерних Android-платформах [2].

Така технологія може підвищити продуктивність ряду Android-додатків, пов'язаних з обробкою зображень, розпізнаванням образів, фізичним моделюванням, клітинно-автоматної моделлю і ін., котрі, в свою чергу, не втратять апаратної незалежності [2].

RenderScript почав підтримуватися в версії Honeycomb/Android 3.0 (API 11). А саме, в Android SDK в директорії платформи-інструментів llvm-rs-cc (автономний компілятор) для компіляції RenderScript (\*.rs файл) в байт-код (\*.bc файл) і генерації Java класів об'єктів (\*.java файли) для структур та глобальних змінних. Основою llvm-rs-cc є Clang — компілятор з невеликими змінами під Android, який являє собою front-end для LLVM компілятора [3].

В ОС Android з'явився фреймворк, побудований на базі LLVM back-end, який відповідає за компіляцію байт-коду під час виконання, лінковку з потрібними бібліотеками, запуск і контроль виконання RenderScript. (рис. 2.) Цей фреймворк складається з наступних частин: libbcc, який займається ініціалізацією LLVM контексту відповідно до зазначених метаданих в байт-коді, компіляцією байт-коду і динамічної лінковкою з потрібними бібліотеками з libRS; libRS містить реалізацію бібліотек (math, time, drawing, ref-counting,...), структур і типів (Script, Type, Element, Allocation, Mesh, various matrices,...) (рис. 3.) [4].

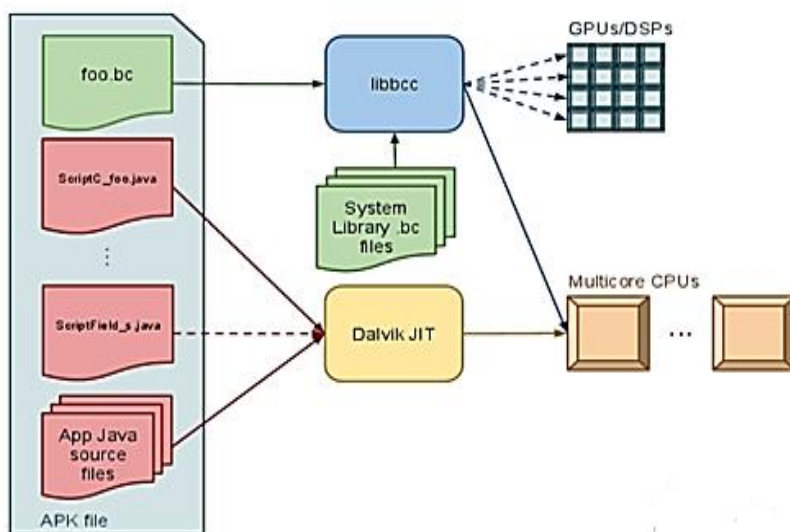


Рис. 2. Схема back-end компіляції Render Script коду

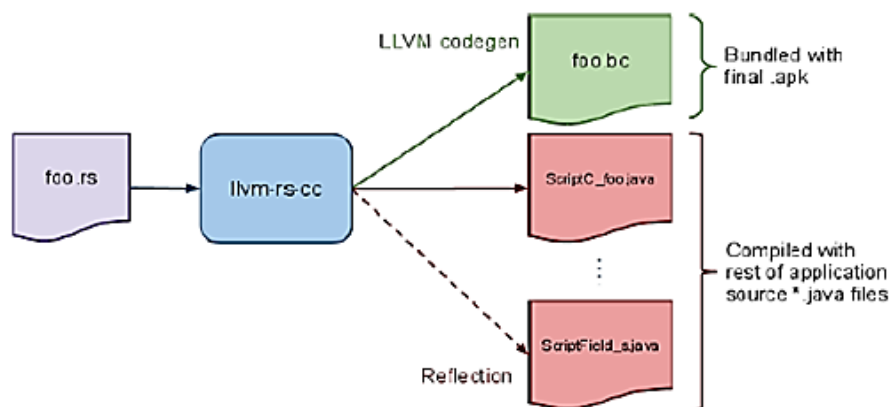


Рис. 3. Схема front-end компіляції Render Script коду

**Постановка задачі.** Метою даної роботи є дослідження компіляторів та мов програмування, які використовуються для створення мобільних додатків для ОС Android. Використовуючи інтегроване середовище розробки Android Studio в роботі створюється модуль, який містить функції, котрі виконують операцію створення монохромного зображення з кольорового та реалізовані засобами мов програмування RenderScript та Java.

Таким чином, проводиться порівняння продуктивності традиційних мови програмування та компілятора з більш низькорівневими засобами, які дають можливості для роботи з апаратним забезпеченням, для редагування зображень.

**Дослідження RenderScript та Java.** В роботі створюється монохромне зображення з кольорового. (рис. 4.) Для цього було написано дві функції: `javaMonoChromeFilter` та `renderScriptMonoChromeFilter`. Перша була реалізована засобами мови програмування Java, а друга — Render Script. Для дослідження використовувався смартфон Meizu Pro 6 з наступними характеристиками:

- Центральний процесор Helio X25: 10 ядер: ARM Cortex-A53 1.4GHz x4 + ARM Cortex-A53 2.0GHz x4 + ARM Cortex-A72 2.5GHz x2
- GPU: Mali-T880
- Hi-Fi ЦАП: Cirrus Logic CS43L36
- Чип NXP Smart PA
- RAM: 4 Гб, LPDDR3

В таблиці 1 наведено результати тестування Android-додатку при різних розмірах оригінального зображення.

Таблиця 1

## Результати тестування програмного модуля

Розмір зображення	960x541	700x394	500x281	300x169
Render Script	156 мс	98 мс	52 мс	30 мс
Java	1083 мс	672 мс	366 мс	150 мс



Рис. 4. Графічний інтерфейс користувача програмного модуля

З таблиці видно, що функція реалізована засобами Render Script працює швидше, ніж функція написана на Java. Поліпшення продуктивності досягається шляхом виконання нативного коду на пристрої. Однак, на відміну від NDK, це рішення кроссплатформенно. Мова розробки Renderscript - це C99 з деякими розширеннями, під час процесу розробки, він компілюється в апаратно-незалежному проміжному форматі і розміщується в пакеті програми. Слід зауважити, що час роботи Render Script-реалізації включає створення Render Script-контексту, виділення і ініціалізацію необхідної пам'яті, створення і прив'язку Render Script до контексту, а також що продуктивність можна підвищити, відмовившись від точності арифметичних операцій з числами, що не принципово для даної задачі.

**Переваги:**

Апаратно-незалежний додаток. Renderscript байт-код, що знаходиться в APK файлі, під час виконання буде скомпільовано в машинний код того апаратно-обчислювального модуля (CPU) платформи, де буде запущений;

Швидкість виконання досягається завдяки паралельним обчисленням;

**Недоліки:**

Відсутність докладної документації для роботи з Renderscript ускладнює розробку додатків.

Відсутність підтримки виконання Renderscript потоків на GPU, DSP. Можливі проблеми під час виконання з балансуванням потоків в гетерогенному запуску, управлінням спільною пам'яттю.

**Висновки.** В результаті проведення дослідження виявлено, що використання Render Script дає певні переваги у продуктивності програмного модуля. У роботі були розглянуті ART(dalvik якщо запускати програму на пристрої з версією

ОС Android < 5.0) і Render Script реалізації монохромної обробки зображень різних розмірів. За рахунок розпаралелювання, роботи компілятора і нативного виконання коду Render Script значно перевершує Java в продуктивності для зображень середніх розмірів. Даний дослід виявляє випадки, коли його застосування може дати підвищення продуктивності додатків, які при цьому залишаються апаратно-незалежними.

#### Список використаних джерел

1. Медникс З., Дорнин Л. Программирование под Android [Текст] / З. Медникс. – СПб.: Питер, 2012. – 460 с.
2. Смартфон. Матеріал з Вікіпедії — вільної енциклопедії [Електронний ресурс]. – Режим доступу: <https://uk.wikipedia.org/wiki/%D0%A1%D0%BC%D0%B0%D1%80%D1%82%D1%84%D0%BE%D0%BD>
3. Казанцев Р. Использование RenderScript на android-устройствах с процессорами Intel [Електронний ресурс]. – Назва з титул. екрана.
4. Марчеті, А. RenderScript: parallel computing on Android, the easy way [Текст] : підручник / А. Марчеті. – Paperback, 2016. – 106 с.
5. Васильев А. Н., Java. Объектно-ориентированное программирование [Текст]: Учебное пособие. – СПб.: Питер, 2011. – 400 с.

#### **Анотація. Кучеренко І.А. Використання Renderscript на Android-пристроях.**

Сьогодні більшість смартфонів і планшетів виробляються на базі Android. Основними причинами поширення цієї операційної системи полягають в наступному.

Android підтримує велику кількість пристроїв від різних виробників. Android характеризується високою доступністю коштів розробки. Інструментальні засоби розробки для Android безкоштовні, в той час як розробка, наприклад, iPhone (від компанії Apple) вимагає значних початкових інвестицій.

Головні переваги портативних комп'ютерів наступному: розмір, тривалість роботи без підзарядки і, звичайно ж, багатофункціональність. Таким чином, ми повинні визнати, що майбутнє ПК - в портативній, легкій та функціональній техніці - планшети, електронні книги, нетбуки і смартфони – і все це в основному працює на операційній системі Android.

**Ключові слова:** Android, Renderscript, Java, компілятори, мова програмування.

#### **Аннотация. Кучеренко И.А. Использование Renderscript на Android-устройствах.**

Сегодня большинство смартфонов и планшетов производятся на базе Android. Основными причинами распространения этой операционной системы заключаются в следующем.

Android поддерживает большое количество устройств от различных производителей. Android характеризуется высокой доступности средств разработки. Инструментальные средства разработки для Android бесплатны, в то время как разработка, например, iPhone (от компании Apple) требует значительных начальных инвестиций.

Главные преимущества портативных компьютеров в следующем: размер, продолжительность работы без подзарядки и, конечно же, многофункциональность. Таким образом, мы должны признать, что будущее ПК - в портативной, легкой и функциональной технике - планшеты, электронные книги, нетбуки и смартфоны - и все это в основном работает на операционной системе Android.

**Ключевые слова:** Android, Renderscript, Java, компиляторы, язык программирования.

#### **Abstract. Kucherenko Igor. Using Renderscript on Android-devices.**

Today, most smartphones and tablets are produced on the basis of Android. The main reasons for the spread of this operating system are as follows.

Android supports a large number of devices from different manufacturers. Android is characterized by high availability of development tools. Development tools for Android are free, while working out, for example, iPhone (from the company Apple) requires considerable initial investments.

The main advantages of portable computers as follows: size, duration without recharging and, of course, multi-functionality. Thus, we should recognize that the future of the PC - in a portable, light and functional technology - tablets, e-books, netbooks and smartphones - all basically just running on the operating system Android.

**Keywords:** Android, RenderScript, Java, compilers, programming language.